REMARKS

No claims have been amended, added, or canceled. Claims 1-24 are pending.

The specification stands objected to. The specification has been amended to address the issue raised in the Office Action. Accordingly, the objection to the specification should be withdrawn.

The Abstract stands objected to. The Abstract has been revised to address the issues raised in the Office Action. Both a marked-up copy as well as a clean copy (on its own page) of the Abstract have been provided. Accordingly, the objection to the Abstract should be withdrawn.

Claims 1-24 stand rejected under 35 U.S.C. 102(b) as being anticipated by Draves (U.S. Patent No. 6,349,355). This rejection is respectfully traversed.

Claim 1 recites, *inter alia,* "accessing said function via said identifier in said privileged and non-privileged modes."

Claim 7 recites, *inter alia,* "logic for accessing said function via said identifier in said privileged and non-privileged modes."

Claim 13 recites, *inter alia,* "means for accessing said function via said identifier in said privileged and non-privileged modes."

Claim 19 recites, *inter alia,* "instructions for accessing said function via said identifier in said privileged and non-privileged modes."

Draves is directed to a computer system in which the user virtual address space is mapped to an offset of the kernel address space. When a kernel function is called, data passed to the kernel function by reference (e.g., via a pointer) is biased to accommodate the offset before dereferencing, thereby permitting the sharing of position dependent code while simultaneously permitting the kernel access to the user memory space.

Draves discloses that on some computer systems with non-segmented memory addressing architecture, such as the MIPS family of processors, the translation lookaside buffer (TLB), a

cache of the mapping between real and virtual addresses stored in the page tables, is software manageable and are indexed by identifiers known as an address space identifiers (ASIDs). That is, each ASID is a pointer to a TLB entry.

Draves discloses associating each process with a pair of ASIDs and TLB entries. A first ASID pointing to a first TLB entry would be used for user mode operations, while a second ASID pointing to a second TLB entry would be used for kernel mode operations. Draves therefore discloses using two TLB entries: for one function to accommodate the addressing differences between the user and kernel modes.

The Office Action asserts at page 5 that "Draves teaches an identifier (e.g., identified by ASID) for accessing the function in both privileged (e.g., privileged) and non-privileged modes (e.g., non-privileged)" and cites to column 11, lines 40-60. Column 11, lines 31-60, which includes the passage cited by the Office Action, is reproduced below:

> In order to implement the virtual memory scheme of FIG. 14, pairs of virtual-to-physical address mappings or entries are maintained for each user process. This is possible because the MIPS processors uses a software-managed TLB--it calls the kernel for TLB manipulations.
>
> Each pair of virtual-to-physical address mappings includes a first address mapping that is identified by a first ASID, and a second address mapping that is identified by a second ASID. Each pair corresponds to a particular user virtual address page. The first ASID and the first entry of the pair, are used when executing a user process in the non-privileged execution mode; the second ASID, and thus the second entry of the pair, are used when executing the kernel from the privileged execution mode.
>
> More specifically, the system call handler of the virtual memory system is configured to switch between a process' first ASID to its second ASID when switching between the non-privileged execution mode and the privileged execution mode. This results in a corresponding switch between the first and second address mappings for a process.
>
> Exemplary address mappings are shown in FIG. 14 for a user process having first and second ASIDs equal to X and X+1. For purposes of illustration, each process is assigned a consecutive pair of ASIDs, including an odd ASID and an even ASID. A first pair of entries 72 for this process includes one entry for ASID X and another for ASID X+1. X is the ASID used in the non-privileged execution mode, and X+1 is the ASID used in the privileged execution mode. In actual implementation, the upper bit of each ASID is used to distinguish between ASIDs of a pair.

Column 11, lines 31-60 (emphasis added)

It is respectfully submitted that the above quoted passage, especially the emphasized portions, demonstrate that Draves discloses the use of a first identifier (i.e., the first ASID) for accessing a function in only the privileged mode as well as a second identifier (i.e., the second ASID) for accessing the same function in the non-privileged mode. As such, Draves fails to disclose, and in fact teach against the above quoted portions of the independent claim, each of which recite using "said identifier" (i.e., a single identifier) for "accessing" said function in "said privileged and non-privileged modes" (emphasis supplied) (i.e., both modes).

Claims 3, 9, 15, and 21 each recite "wherein the data structure is a table that maps identifiers to functions." As noted above, the data structure associated with each identifier in Draves is the TLB. Since TLBs map between real addresses and virtual addresses, and not between identifiers and functions, Draves fails to disclose or suggest a "table that maps identifiers to functions," as recited in each of independent claims 3, 9, 15, and 21.

This is contrary to the Office Action's assertion on page 5, which cites to column 7, line 32, column 8, line 43, and column 11, line 4. It is respectfully asserted that the Office Action is in error. Each passage relied upon by the Office Action for support is reproduced below:

i. Column 7, line 32 is a sentence fragment. However, column 7, lines 32-34 is the entire sentence and states: "FIG. 8 shows the same virtual memory system 45 as shown in FIG. 7, but also shows the virtual-to-physical address mappings 50 that define the virtual address space.

ii. Column 8, line 43 is a sentence fragment. However, column 8, lines 43-44 is the entire sentence and states: "Refer now to FIG. 9, which shows a user virtual address space, a system virtual address space, and physical memory."

iii. Column 11, line 4 states: "The MIPS processor does not use segmented addressing."

The passages relied upon by the Office Action merely refer to illustrations of mappings between two address spaces, and indicate that the MIPS processor does not use segmented addressing. In fact, they do not disclose or suggest the use of any table that maps identifiers to functions. As previously noted, a TLB merely maps a first address space to a second address space. Draves discloses the use of ASIDs, which as previously noted is a pointer to an entry of a TLB. An ASID is an identifier, but it is not a part of the TLB itself. Thus, what Draves discloses is an

identifier for pointing to an entry in a table which maps from one address space to another address space. This does not read upon claims 3, 9, 15, and 21, as each of these claims recite a "table that maps identifiers to functions."

Accordingly, claims 1, 3, 7, 9, 13, 15, 19, and 21 are believed to be allowable over the prior art of record. The depending claims (i.e., claims 2, 4-6, 8, 10-12, 14, 16-18, 20, and 22-24) believed to be allowable for at least the same reasons as the independent claims.

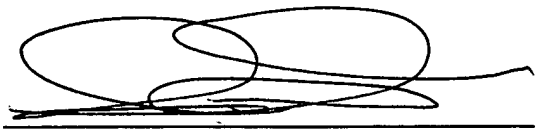
CONCLUSION

In light of the amendments contained herein, Applicants submit that the application is in condition for allowance, for which early action is requested.

Please charge any fees or overpayments that may be due with this response to Deposit Account No. 17-0026.

Respectfully submitted,

Dated: March 13, 2006

By: 

Christopher S. Chow
Reg. No. 46,493
(858) 845-3249

QUALCOMM Incorporated
Attn: Patent Department
5775 Morehouse Drive
San Diego, California 92121-1714
Telephone: (858) 658-5787
Facsimile: (858) 658-2502

AMENDMENT TO THE ABSTRACT

Please amend the Abstract as indicated below. A clean copy of the Abstract appears on the following page.

~~[0042]~~ Methods and apparatuses are provided ~~System~~ for dynamic registration of privileged mode hooks in a device[[.]] that can operate in a ~~The system includes a method for dynamically registering a function in a device that includes at least two operating modes comprising a~~ privileged mode and a non-privileged mode. A data structure is provided which maps between identifiers and functions. An ~~The method comprises identifying an~~ available slot in the [[a]] data structure is used to store ~~that maps identifiers to functions, and storing~~ a pointer associated with a ~~the~~ function. ~~in the slot.~~ The identifier can then be made accessible to non-privileged applications. ~~method also comprises retrieving an identifier that is associated with the slot, and making the identifier accessible to non-privileged applications.~~